

SYSRPC - Stub Generation

The Stub Generation function is used to generate client stub subprograms. Though stubs are actually not required if automatic Natural RPC execution is used, it can be advantageous to generate them anyway.

For more details, see Stubs and Automatic RPC Execution in the section Operating a Natural RPC Environment in the Natural RPC documentation.

This section covers the following topics:

- Invoking Stub Generation
 - Parameter Specification
 - Example of Stub Generation
-

Invoking Stub Generation

Attention:

The Stub Generation function invokes the Natural editor. As a result, data stored in the source work area can be lost when invoking Stub Generation. A corresponding message will warn you not to delete any existing entries unintentionally: choose PF12 to cancel the function or choose ENTER to confirm the action and clear the source work area.

The stubs are generated in the current library. We strongly recommend that you log on to the application library (or one of its steplib) used by the client at execution time of the remote CALLNAT.

Invoking and using the Stub Generation function

1. In the Code field of the Client Maintenance screen, enter **SG**.
The Generate Client Stub Routine window appears.
2. Enter the name of the stub subprogram to be generated.
The name of the stub subprogram must be the same as the name of the remote CALLNAT program.
The name of the library is preset with the name of the current library and cannot be changed.
3. Choose Compression Type 0, 1 or 2 (default is 1); see Using Compression as described in Operating a Natural RPC Environment in the Natural RPC documentation.
4. Choose ENTER.
 - If the subprogram specified in Step 2 already exists in the library assigned, a corresponding window is displayed:

Specify **N** (No) and choose ENTER if you do not want to generate a new stub.
You will be returned to the Client Maintenance screen.

Specify **Y** (Yes) and choose ENTER.
The parameter data area of the existing subprogram is displayed in the Stub Generation screen.
 - If the subprogram specified in Step 2 does not exist, an empty Stub Generation screen is displayed.
5. Add or modify the parameters to be used in the stub subprogram: see Parameter Specification below.
On the Stub Generation screen, the same editor and line commands apply that are valid for the Service Directory Maintenance function (see the relevant section).
6. Choose ENTER to generate the stub subprogram and to exit.
The stub subprogram is generated in the library assigned in Step 2.

A window appears which indicates the size the stub requires for sending data from the client to the server or vice versa. The size includes internal RPC information used for the stub. The indication of the size helps you configure the middleware layer used, for example, the EntireX Broker attribute file.

Below is the text that appears in the window when you generate a stub from the example subprogram TESTS5 (see Example of Stub Generation below):

```
Stub TESTS5 is generated in library TEST
  It requires:
    Send length: 2249 bytes
    Receive length: 2221 bytes
```

If the Send or Receive length exceeds the Entire Net-Work limit of 32000 bytes, a window appears with a corresponding warning. Enter **Y** (Yes) to continue, or **N** (No) to cancel the generation.

If you choose **Y**, this setting is kept for the entire SYSRPC session, that is, you can continue generating stubs without receiving further warnings.

If the total data (without internal RPC information) sent or received exceeds the limit of 1073733630 bytes (which is 1 GB minus 8 KB of internal RPC information), SYSRPC stops processing and issues a corresponding error message. This error message displays the subtotal of the data in bytes that could be transferred at the field up to which the subtotal was calculated. The corresponding field is then marked. In this case, reduce the amount of data and then continue generating the stub.

If the stub was generated in the library SYSRPC, you must move the stub to the application library or steplib using the appropriate Natural transfer utility (SYSMAIN, SYSTRANS or Natural Object Handler). Note that you may have to recatalog the stub sources in the target environment.

Parameter Specification

In the input fields provided on the Stub Generation screen, you can specify the parameters that are used in the stub subprogram:

Field	Description
Attr	The attribute which specifies the parameter as: M (modifiable field), O (output field) or I (input field).
Type	The Natural data type, such as N (numeric). Data Types C and Handle are not allowed.
Len	The length of the variable. Natural Data Type A is restricted to 1073733630 bytes, Data Type B is restricted to 536866815 bytes. Dynamic variables are not allowed.
Prec	Only applies to data types N (numeric) and P (packed). Optional. The precision of the variable, that is, the number of digits after the decimal point.
Dim1/2/3	Only applies to arrays. Optional. The first, second and third dimension of the variable. The maximum of occurrences is 1073733630 bytes for each dimension.

Example of Stub Generation

The following example shows four modifiable parameters that correspond to the variable definitions in the Natural subprogram TEST5 supplied in the Natural system library SYSRPC:

```

DEFINE DATA
PARAMETER
  2 #IDENTIFIER  (A10)
  2 #N-OF-ID     (I4)
  2 #FREQ        (P5.2)
  2 #A100        (A100/5,4)

```

Stub Generation							
	Attr	Type	Len	Prec	Dim1	Dim2	Dim3
1	M	A	10				
2	M	I	4				
3	M	P	5	2			
4	M	A	100		5	4	